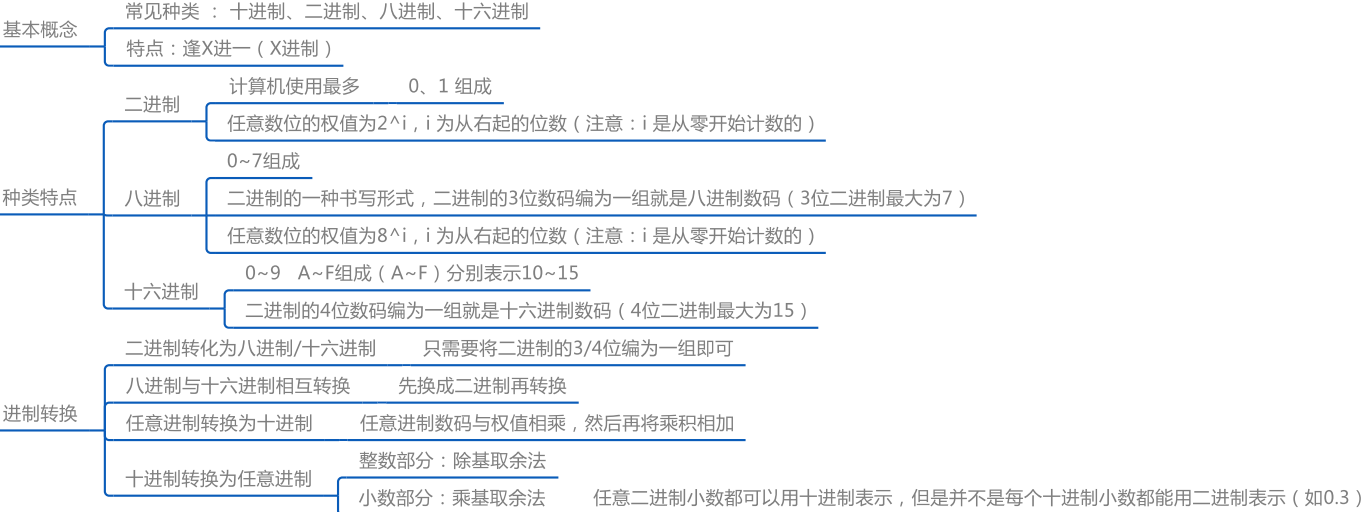


# 2.1数制与编码 (上)

## 2.1.1 进位计数制及其相互转换



## 2.1.2 真值和机器数

真值：带符号的 如-1，2，-3，4

机器数：常用最高位 0表示正 1表示负

2.1数制与编码（下）

2.1.3 BCD码

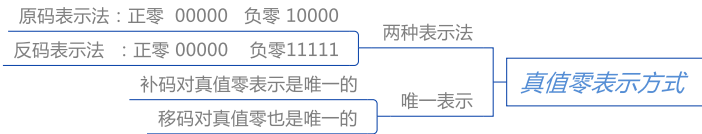
|            |  |
|------------|--|
| 8421码（最常用） | 注意：如果两个8421码相加后的和小于9（10进制），需要加6（10进制）进行修正                    |
| 余3码（无权码）   | 在8421码的基础上加（0011），因为每个数都多3，所以称为余3码<br>8--->1011              |
| 2421码（有权码） | 权值由高到低为 2, 4, 2, 1<br>>=5的4位二进制数中最高位为1，<5的最高位为0<br>5--->1011 |

2.1.4 字符与字符串

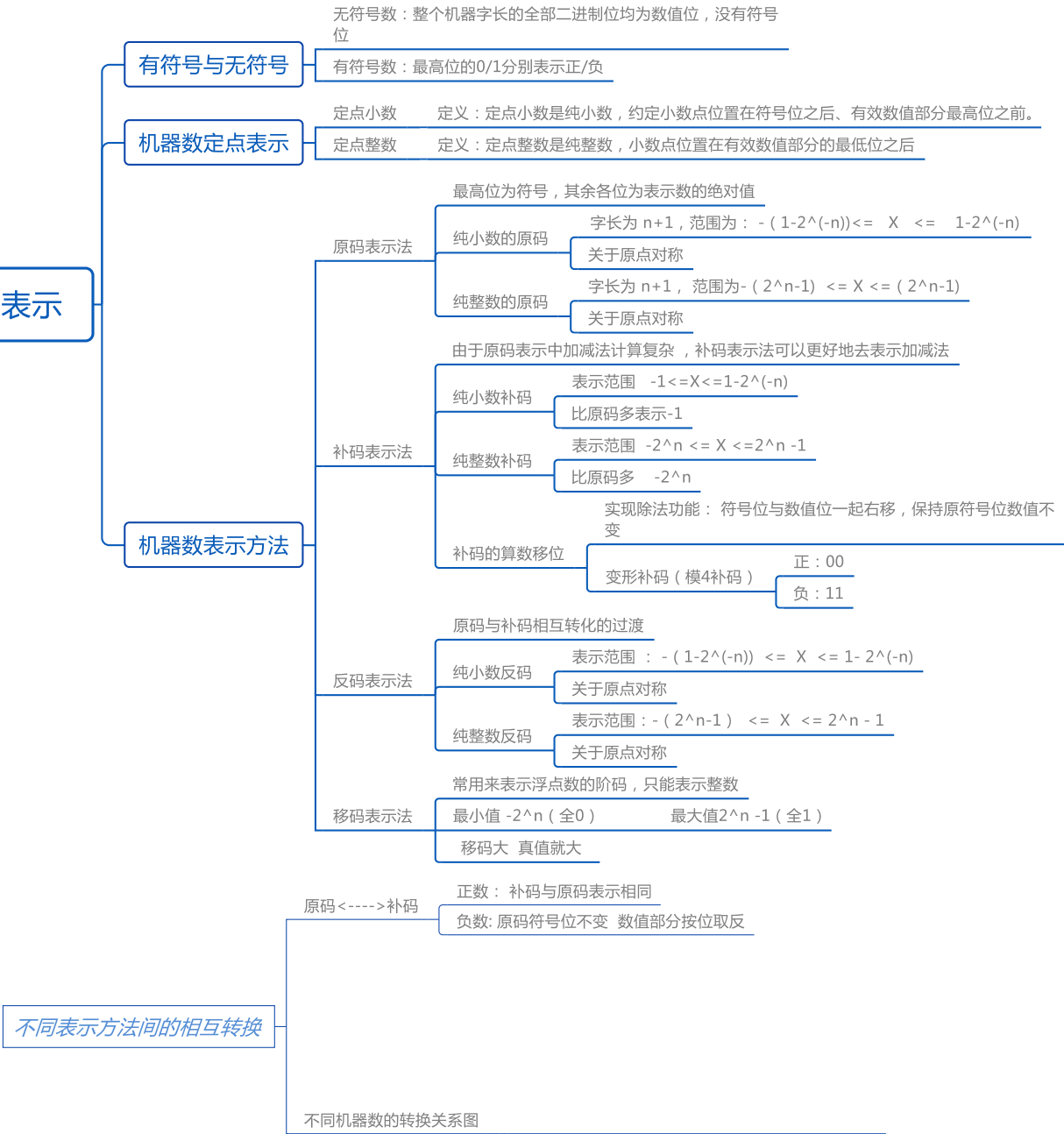
|                             |  |
|-----------------------------|--|
| 字符串编码ASCII码                 | 7位二进制编码  |
| 汉字的表示和编码                    | 每个编码用两个字节表示  |
| 种类                          | 输入编码 计算机输入<br>汉字内码 计算机内部处理<br>汉字字形码 计算机输出            |
| 字符串存放                       | 小端模式 将数据的最高有效字节存放在高地址单元中<br>大端模式 将数据的最高有效字节存放在低地址单元中 |
| 从低地址到高地址逐字符存储，常采用'\0'作为结尾标志 |  |

2.1.5 校验码

|                         |   |
|-------------------------|---|
| 概念：能够发现或者自动纠错的数据编码      |   |
| 原理：通过添加一些冗余码，实现检验或者纠错编码 |   |
| 奇偶校验码                   | 奇校验码：有效信息位和校验位中 1 的个数为奇数<br>偶校验码：有效信息位和校验位中 1 的个数为偶数<br>码距：2  |
| 但是如果编码中出现偶数位错误，无法检测     |   |
| 海明码                     | 在有效信息位中添加几个校验码形成海明码<br>不仅可以发现错位，还可以对错位进行纠错<br>编码最小码距L越大，检测位数越多，纠错能力越强（纠错能力恒小于等于检测能力）<br>先检错，才能纠错    |
| 补充                      | 海明码有1位纠错，2位检错能力<br>为了区分1位错和2位错，还需添加"全校验位"对整体进行偶校验<br>注意：有的题目位置编号可能是从小到大的，但处理方法雷同                    |
| CRC（循环冗余码）              | 常用于大量的数据传送时的校验<br>接收到循环冗余码后，对生成多项式做模2除法，余数为0则无错误<br>余数不为0，对相应位置取反                                   |
| 检错、纠错能力                 | 可检测出所有奇数个错误<br>可检测出所有双比特的错误<br>可检测出所有小于等于校验位长度的连续错误<br>若选择合适的生成多项式，且 $2^R \geq K + R + 1$ ，则可纠正单比特错 |



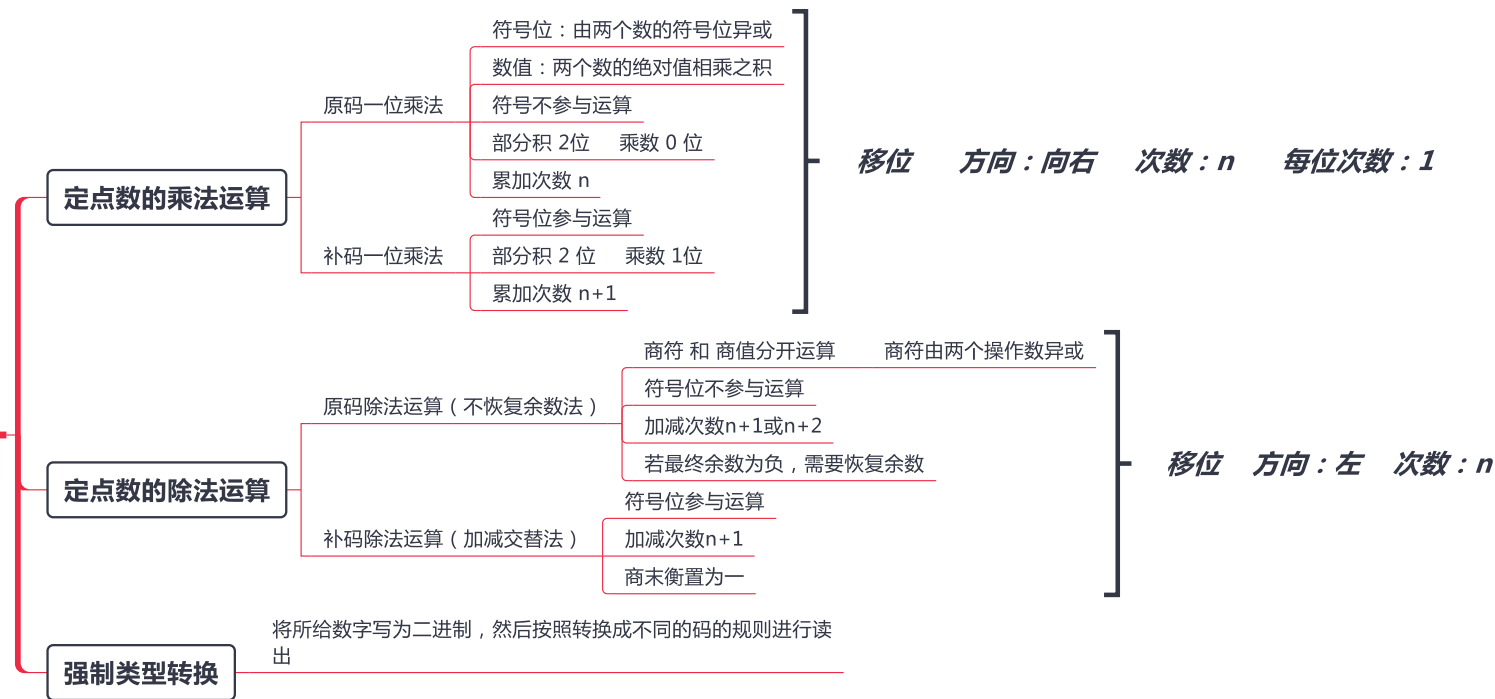
## 2.2.1 定点数的表示



## 2.2.2 定点数的运算（上）



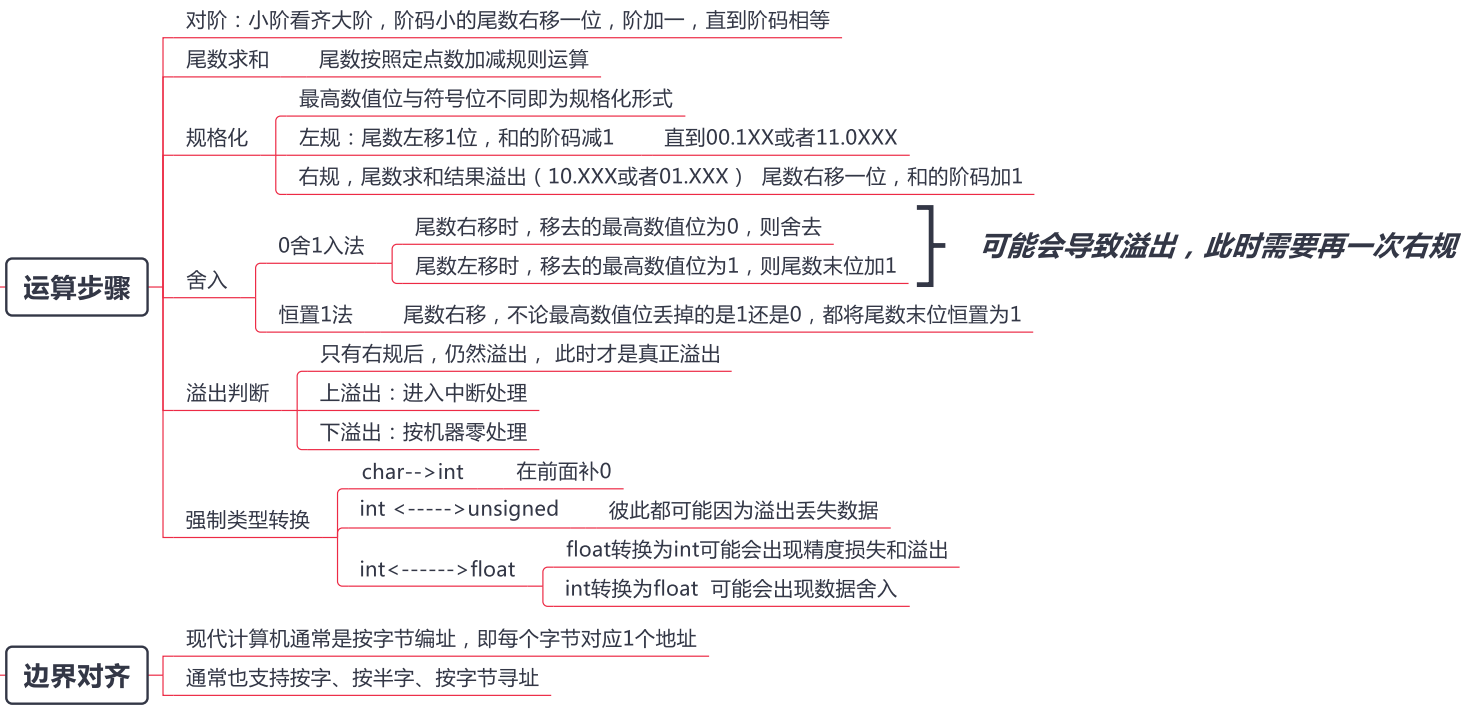
## 2.2.2 定点数的运算（下）



# 2.3.1浮点数的表示



# 2.3.2 浮点数的加减运算



2.4算术逻辑单元（ALU）

运算器组成：算术逻辑单元 累加器 状态寄存器 通用寄存器组

2.4.1串行加法器和并行加法器

- 一位全加器
  - 两个加数输入，以及低位进入输入
  - 本位结果和进位输出
- 串行加法器
  - 只有一个全加器，数据逐位的送入加法器中运算，逐位送回寄存器
  - 操作数n位 则进行n次
  - 成本低，但是速度慢
- 并行加法器
  - 多个加法器共同组成，每个全加器都有一个低位送来的进位输入，向高位的进位输出
  - 进位方式
    - 串行进位 将全加器串接在一起，每级进位依赖于前一级进位
    - 并行进位 同时进位，各级进位信号同时形成

提高并行加法器速度的关键在于加快进位产生和传递速度

2.4.2 算术逻辑单元的功能和结构

ALU：算数运算与逻辑运算